Multi-Node Localization and Identity Estimation Based on Multi-Beacon Searching Algorithm

Mofeed T. Rashid, Abdulmuttalib T. Rashid, Ammar A. Aldair

Electrical Engineering Department; University of Basrah; Basra; Iraq; e-mail: mofid76@gmail.com

Corresponding author: mofeed.rashid@uobasrah.edu.iq

The accuracy of multi-nodes localization and identity estimation algorithms directly affects the performance of multi-agent systems like wireless sensor network (WSN), multi-robot, cellular phone and so on. In this paper, a novel algorithm is introduced in order to achieve a high accuracy for multi-node localization and identity estimation. A grid of reference beacons is distributed uniformly in the environment, where each beacon has a light-emitting diode (LED) to emit the light to four light dependent resistor (LDR) sensors that are equipped on each node. The localization process is achieved based on three proposed algorithms. Firstly, a modified binary search algorithm is utilized to estimate the approximate location of the node by a group of neighbor LEDs. Secondly, the accurate localization algorithm is used to find the accurate location of each node by reducing the number of neighbor LEDs. Finally, two algorithms are introduced to evaluate the location and identification of each node: the centroid algorithm and the minimum bounded circle algorithm. In the minimum bounded circle algorithm, a new algorithm called the "maxima boundaries convex hull algorithm" for polygon convex hull construction is introduced. Several simulation processes have been implemented for testing the proposed algorithms. The obtained results show that the proposed algorithms have a considerable performance in estimating the accurate localization of the nodes.

KEYWORDS: Localization, Multi-node system, Binary search algorithm, Minimum bounded circle algorithm, Centroid algorithm.

1. Introduction

Recently, the rapid development of multi-agent systems such as WSN, multi-robot systems, and cellular phones has led to the development of the localization algorithms. This development plays an influential role in the performance of these systems where the localization accuracy and execution time are the main quality parameters of these algorithms.

Localization is a term that refers to the finding mechanism of objects spatially in its environment, such as in multi-robot systems [22, 27]. However, in a WSN, localization is the process of computing the locations of sensor nodes with respect to the WSN environment [7]. The nodes localization systems are categorized into two systems: the single-node system and the multi-node systems. The multi-node system has several superiorities over single-node systems for localizing themselves more efficiently [14], improving the speed of search and performing the exploration missions [18].

The field of indoor multi-nodes localization algorithms has been deeply studied by researchers. These algorithms are categorized according to sensors that are used to estimate the location of the nodes. Many different types of sensors are used to perform this mission such as vision system, laser range finder sensor (LRF), technologies of Bluetooth, radio frequency identification (RFID), ultrasonic range sensor technology, Wi-Fi wireless network, and infrared light emitting diode (IR-LED) detectors. A vision system of position sensing has an influential performance for localizing of multi-nodes and identifying the nodes resided in an indoor environment [11]. However, this method needs a complex software to increase the accuracy and overcoming environment limitation issues. The "time of flight" concept based sensors, such as Ultrasonic [26] and LRF [25] localization systems, are characterized by a simple hardware structure with high localization accuracy. However, these types of sensors are not accurate if they are neighbors for unknown moving objects and it is difficult to identify the nodes. For LRF, a transparent wall technique is used for the indoor environment. Bluetooth [5] and Wi-Fi [8] devices are relatively easy to install to the mobile nodes, but there are problems with accurate localization due to the interference of other signals with them. Many IC tags should

be employed for accurate localization by the RFID system. In addition, these tags need for optimal distribution in order to achieve localization tasks [19]. Therefore, this technology requires a notable deal of preparatory work such as building a map of identifiers and RFID tag locations. In addition, RFID is relatively expensive.

Although the IR-LEDs need optimal distribution, the VLC and IR systems are characterized by a fast response, good stability, high information security, and the impact of environmental changes on them are minimal. Due to these advantages, those systems are used in indoor localization of multi-nodes. Almost all the localization applications that employed the IR system depend on the identified artificial tags. These tags are classified into passive and active tags. The passive tags based localization implemented by distributing tags is able to reflect IR signals, so they do not need for power. The IR camera and IR detectors, which are fixed on the nodes, sense the IR signal that emitted by nodes and reflected by tags [21, 13]. IR emitter is usually used as an active tag and optimally distributed in the environment. The emitted IR signals are sensed by the IR detectors in order to estimate the locations of nodes [3, 10]. In [24], Wang and Takahashi mentioned that the IR system based localization needs a unique ID to encode each tag. This led to limiting the number of nodes that can be localized in the indoor environment. However, the cost is an important limiting factor for extended flexibility. As it is known, most of the multi-node systems, such as robotics, WSN and so on, work in large open areas such as large stores, large museums, stadiums, health institutions, etc. Therefore, they need a large number of tags. As a consequence, there is a great demand for a multi-node localization system that is characterized by a suitable cost for a large number of tags and is able to face the challenges of the complexity of the system.

With the rapid increase of smart technology, the researchers are seeking a sensor that provides precise distance detection taking into account the cost. The features of Light Dependent Resistor (LDR) detectors, such as accuracy, low price, and long lifetime, led us to use this system in localization scenarios. Moreover, there is no overlap problems compared with IR-LED and Laser sources that reduce noise effects.

Many localization techniques have been developed for multi-nodes in indoor environments. Comparisons have been introduced for indoor multi-nodes localization techniques in [4, 9, 15]. A survey for approaches employed the IR system for node localization is introduced to create an idea for the reader about the contribution made in this research.

The Hagisonic StarGazer localization system is presented in [23] where the overall cost has been evaluated as 1750 $ for an environmental area of about 2000 m². This cost includes the price of passive tags (HLD1-L 4 x 4 grid), tags installation on the ceiling, in addition to the cost of localization sensors. Due to the high cost of localization sensors, this system is unsuitable for the multi-nodes operate together within unknown environment, which need for unique code for each node. This challenge led to increasing the system complexity and cost. However, our proposed system will solve this problem by increasing the number of nodes localized at the same time, reducing the cost as well as avoiding the complexity of the system.

The majority of approaches of indoor localization are employing active tags such as IR-LED for emitting signals, while IR detectors used to receive these signals as IR beacons that contain information about tag only [21]. Their approach includes placing beacons that have known locations in the environment which then is detected by using scanners fixed on the nodes. Sixteen receivers were used for this purpose, they were distributed as a circle covering the 360° range. In addition, the Extended Kalman filter (EKF) has been used for estimating the location of each node. In our approach, four LDR detectors will only be used in order to reduce the cost without losing good accuracy and so there is no need for known the initial position of the nodes.

In [13], Lee and Song employed unique IDs for encoding IR LED as artificial tags for nodes localization. Tags placed on the ceiling in which dividing into zones of tags, for each active node there is a unique identified sector in order to know the node location within the identified zones. At each node, there is a receiver for enabling the estimation of the position of the node during its motion. They introduced an algorithm to include both distance measurement and IR ID information so that the size of the uncertainty can be reduced. However, the starting point must be identified, and unit length is required, while the overlapping cannot

be analyzed. Our Proposed system is LEDs that will be distributed on the ground while LDR is fixed on the robot body and close to the LEDs. Thus, there would be a small chance of overlapping. Our proposed system is LEDs will distributing on the ground while LDR is fixed on the robot body and close to the LEDs, so, there is a very low chance of overlapping.

Some researchers employ the receiver as an active tag in localization systems such as Gorostiza et al. [6], and they developed these systems. In these systems, the active tags fixed on the node while the sensors distributed in the environment and received a sinusoidal modulated IR signal transmitted by the node. One emitter set is placed on each node while there is one photo-detector for each receiver fixed in the environment (ceiling). Range measurement based on phase shift has been used to measure different distances, and then hyperbolic trilateration used to estimating node location. Using these systems will avoid the challenges of ID coding. Nevertheless, it requires a controller that sends back the information of the estimated position of nodes in the environment to each node. This process will increase the complexity of the system, furthermore cost and time, but the proposed system introduces a method for evaluating node position locally by each node, so, there is no need for an additional controller for retransmitting the estimated location of the node.

This paper aims at introducing a multi-node localization and identification system based on an LED-LDR system which is supported by a multi-node searching algorithm. This system achieves node location estimation and node identification with respect to a grid of reference beacons in the indoor environment. A grid of LEDs distributed uniformly in the global indoor environment represents these beacons. Four LDR sensors are used to sense the lighting LEDs according to a new proposed searching algorithm equip each node. During the scan process, the LDR sensors for each node collect its reading and send the information to the computer transmitter/receiver board as shown in Figure 1. For example, using an NRF24L01 transmitter/receiver unit to wirelessly transfer data between the computer in one side and the node on the other side. The localization process is achieved in three stages. In the first one, the modified binary search algorithm is used to estimate the approximate location of the node based on a group of neighbor LEDs. In the

second stage, an accurate location of each node is addressed through a new proposed searching algorithm. This algorithm is used to estimate a more accurate location of the node by reducing the number of neighbor LEDs of the node. In the last stage, two algorithms are used to evaluate the location and identification of the nodes: the centroid approach and the minimum bounded circle algorithm. In the minimum bounded circle algorithm, Chan's algorithm is used to draw a circle containing all the set points, which represent a convex hull polygon. In addition, a new proposed algorithm to enhance the execution time has replaced Chan's algorithm.
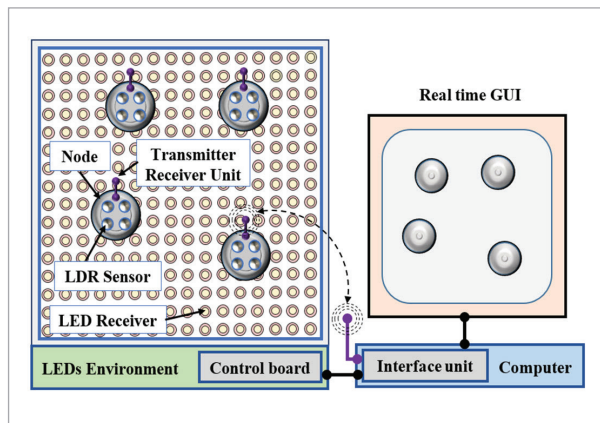
2. The Multi-Beacon Searching Algorithm

In this section, the background of the proposed multi-node localization system and its modification will be explained in detail. In addition, the hardware structure of the proposed LED-LDR system is described in detail. As mentioned in the introduction, the overall implementation of the proposed system is divided into three stages, so, the description of these stages is shown in the following subsections.

The proposed algorithms discussed in detail in Section 2. Section 3 describes the simulation results of the Multi-Beacon Searching Algorithm. Finally, Section 4 draws the conclusions of the paper.

Figure 1
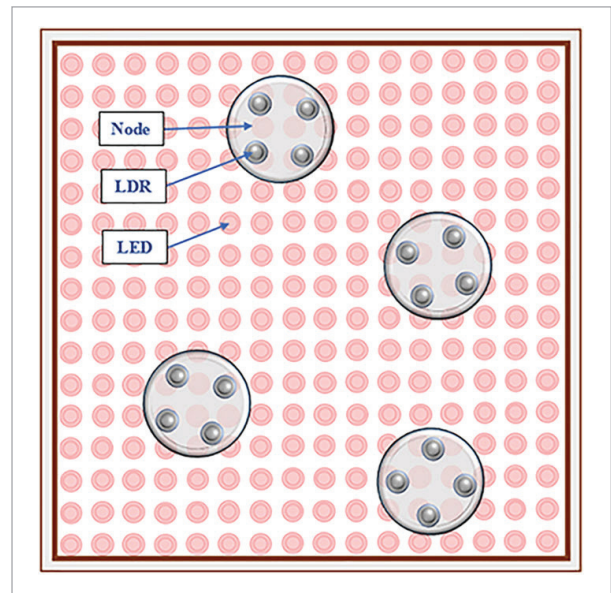Scheme of the nodes localization system

2.1. Approximate Localization Algorithm

This section introduces the first stage of the proposed system. The approximated localized algorithm is used to estimate the approximate location of the node by a group of neighbor LEDs. The localization process depends on using a grid of LEDs that are distributed uniformly in the known environment as shown in Figure 2. The absolute location of each LED is computed according to its address in the LED grid. The LDR sensors are placed on the nodes to sense the status of LEDs in the environment and provide the approximate location of the nodes. Actually, this algorithm depends on turning on and off the LEDs sequentially according to the modified binary search algorithm. This algorithm is a two-dimensional binary search algorithm that replaces the decimal numbers by the digital data representing the status of lighting LEDs.

Figure 2
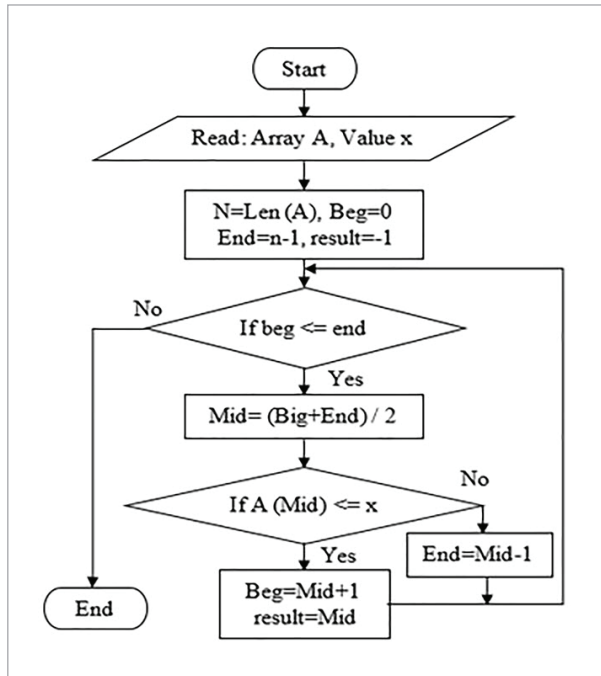Localization environment with 16 x 16 LEDs

2.1.1. Binary Search Algorithm

The binary search algorithm is a simple logarithmic algorithm used with sorted numbers [20]. The array of numbers searched is divided into half repeatedly, and the search value is checked in each step. If the searched value is less than the middle element in the

array, then this array reduced to the lower half else-where to the upper half. This process is repeated until the searched value found or the array is empty. Figure 3 shows the flowchart of the binary search algorithm.

Figure 3
Flowchart of the Binary search algorithm

2.1.2. Modified Binary Search Algorithm

The last subsection shows that the binary search algorithm is designed to search for one number among several numbers arranged in a one-dimensional array. In our system, there is more than one value that should be searched, which represented by the LDR sensors, and instead of a one-dimensional array of numbers, the system has a two-dimensional array of logical values that represent the status of the LEDs. Therefore, the binary search algorithm must be modified to be suitable for use in the proposed system. In this section, two localization algorithms have been suggested: single node and multi-node modified binary search algorithms. Both algorithms are used to estimate the approximate location of the nodes depending on the two-dimensional binary searching in the LEDs grid.

A. Single node modified binary search algorithm

This algorithm is used to estimate the approximate location of a single node by labeling the entire neighbor LEDs to the node location according to the following steps:

Step 1: Estimate the approximate columns of the neighbor LEDs.

In this step, the readings of the LDR sensors are used to distinguish the neighbor columns to the node location. The modified binary search algorithm was implemented on the columns of the LED grid. Firstly, half of these columns are turned on and the rest are turned off, then the statuses of the LDR sensors are checked. Figure 4 shows several cases that may be obtained depending on the node location as described below.

1 All of the LDR sensors may sense the light of LEDs as shown in Figure 4(a). This case occurs when the position of node is on the right part of the environment. This means that the left part of the environment has no node. In this case, the LEDs located in the right half of the right part should be turned on and the rest should be turned off. This process is repeated until one of the following cases occurs.

2 The single LDR sensor may senses the light of LEDs as shown in Figure 4(b). Let the distance between each two neighbor LEDs is D pixels, the distance between each two-neighbor LDR sensors is R pixels and the sensing range of the LDR sensor is S pixels. In this case, the worst probability is shown in Figures 4(a)-(b). The value of Lt is computed as

$$L_t = N + S, \tag{1}$$

where N is the distance between the middle lighted column and the left far LDR sensor. The number of left approximate columns is computed from the following equation:

$$Lcount = L_t / D. \tag{2}$$

The value of R_t, shown in Figure 4(a), is computed as
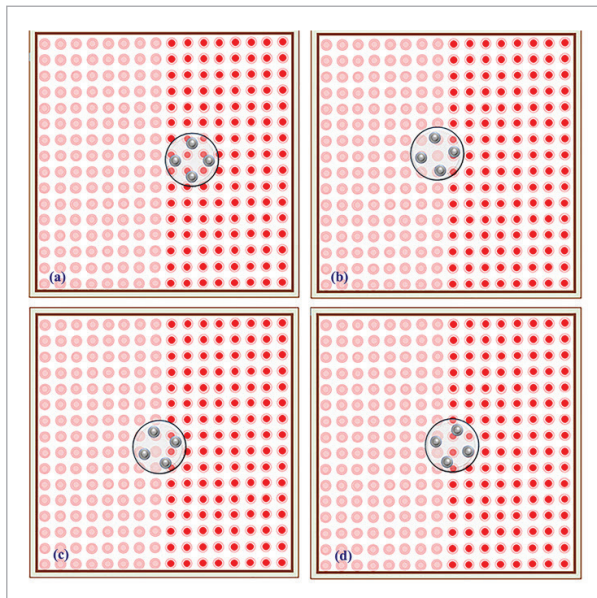
$$R_t = M + S, \tag{3}$$

where M is the distance between the middle lighted column and the right far LDR sensor. Therefore, the number of right approximate columns is computed as:

$$Rcount = R_t / D. \tag{4}$$

3 Two LDR sensors may sense the light of the LEDs as shown in Figure 4(c). The worst-case for node location is to have two LDR sensors sensing an LED light is in Figures 5(c)-(d). The process of the approximation columns estimation of the neighbor LEDs is similar to the procedures that are used to estimate the location of the single LDR sensor as described above.

Figure 4

Cases for the node location on the lighted columns depend on the sensitivity of the LDR sensors. (a) All LDRs sensing (b) Single LDR sensing (c) Two LDRs sensing (d) Three LDRs sensing
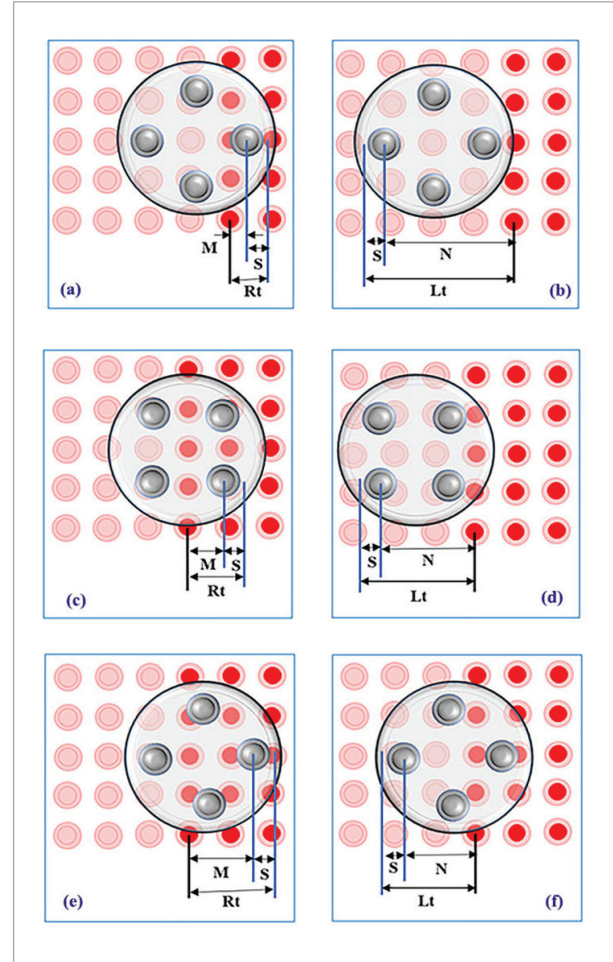
Figure 5

The worst probability for node location dependent on the sensitivity of the LDR sensors. (a, b) single LDR sensing (c, d) two LDRs sensing (e, f) three LDRs sensing

4 Three LDR sensors may sense the light of LEDs as shown in Figure 4(d). The worst probability for node location to make three LDR sensors sense the LED light as shown in Figures 5(e)-(f). The same procedures that are described in Step 2 can be used to estimate the location of the node.

5 All of the LDR sensors may not sense the light of the LEDs. This case occurs when the node position is on the left part of the environment. This means that the right part of the environment has no node. In this case, the LED columns of the right half of the left part should be turned on and the rest should be turned off.

This process is repeated until one of the last three cases occurs.

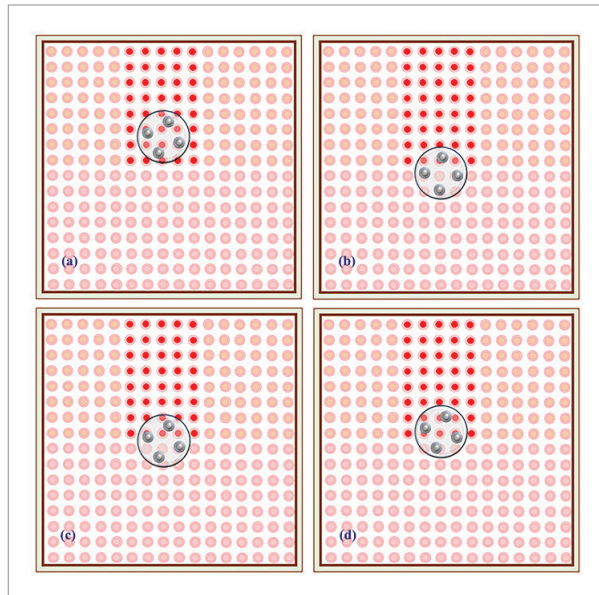**Step 2:** Estimate the approximate rows of the neighbor LEDs.

In this step, the readings of the LDR sensors are used to distinguish the neighbor rows to the node location. The same procedures in the last step of implementing the modified binary search algorithm on the columns are repeated in this case. In the beginning, half of these rows are turned on and the others are turned off, then the status of the LDR sensors is checked. Figure 6 shows the cases which may be obtained from the

rows configuration depending on the node location.

1   Figure 6(a), shows the case where all of the LDR sensors sense the light of LEDs. This case occurs when the node position is on the upper part of the environment. This means that the lower part of the environment has no node. In this case, the LED rows of the upper half of the upper part are turned on and the rest are turned off. This process is repeated until one of the following cases occurs.

Figure 6

The node location cases on the lighted rows dependent on the sensitivity of the LDRs. (a) All LDRs sensing (b) single LDR sensing (c) two LDRs sensing (d) three LDRs sensing

2   Figure 6(b) shows the case where single LDR sensor sense the light of the LEDs. Here, the worst case is shown in Figures 7(a)-(b). The value of L_o is computed from the following equation:

$$L_o = B + S, \tag{5}$$

where B is the distance between the middle lighted row and the lower far LDR sensor. The number of lower approximate rows is given by:

$$Locount = L_o / D. \tag{6}$$

The value of U_p is computed from the following equation:

$$U_p = C + S, \tag{7}$$

where C is the distance between the middle lighted row and the upper far LDR sensor. The number of upper approximate columns is given by:
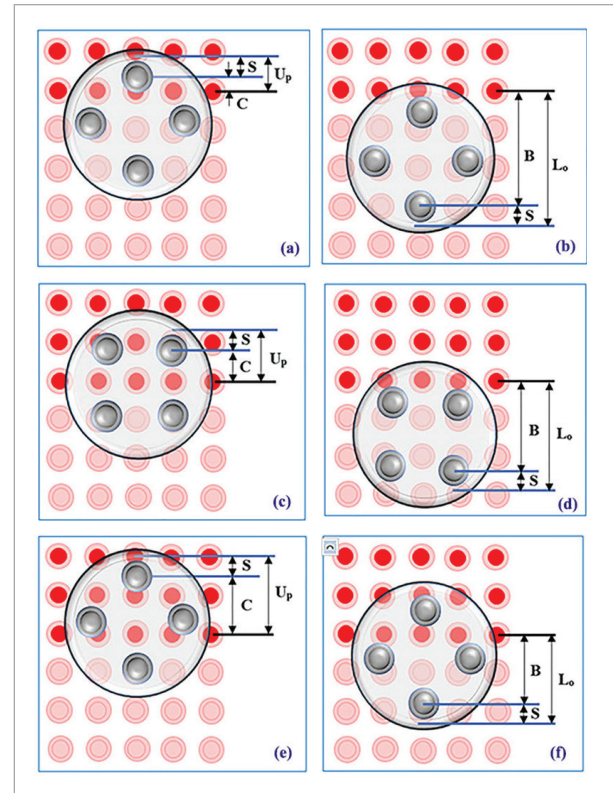
$$Upcount = U_p / D. \tag{8}$$

3   Figure 6(c) shows the case where two LDR sensors sense the light of LEDs. The worst case for node location to make two LDR sense the LED light is shown in Figures 7(c)-(d). The same procedures in Step 2 are used to estimate the location in this case.

4   Figure 6(d) shows the case where three LDR sensors sense the light of the LEDs. The worst case for node location to make three LDR sense the LED light is shown in Figures 7(e)-(f). The same procedures of Step 2 can be used in this case to estimate the location of the node.

Figure 7

The worst probability for node location dependent on the sensitivity of the LDR sensors. (a, b) single LDR sensing (c, d) two LDRs sensing (e, f) three LDRs sensing

5 All of the LDR sensors may not sense the light of the LEDs. This case occurs when the node position is at the lower part of the environment. This means that the upper part of the environment has no node. In this case, the LED rows of the upper half of the lower part are turned on and the rest are turned off. This process is repeated until one of the last three cases occurs.

B. Multi-node modified binary search algorithm

This algorithm is used to estimate the approximate location of multi-node by labelling all the neighbor LEDs to the nodes location according to the following steps.

Step 1: Estimate the approximate columns for each node.

In this step, the readings of the LDR sensors are used to distinguish the neighbor columns to each node location.

The modified binary search algorithm is implemented on the columns of the LED grid. In the beginning, half of these columns are turned on and the others are turned off then the status of the LDR sensors is checked. Figure 8 shows several cases that may be obtained depending on the nodes locations.

1 All of the LDR sensors of all the nodes may sense the light of the LEDs. This case occurs when all the nodes positions are on the right part of the environment. This means that the left part of the environment has no node. In this case, the LED columns of the right half of the right part are turned on and the rest are turned off. This process is repeated until one of the following cases occurs.
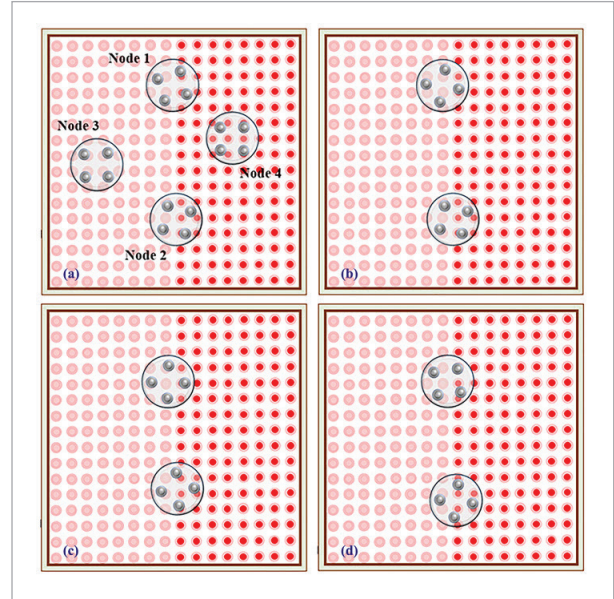
2 One, two or three LDR sensors for one node only sense the light of the LEDs and the other nodes locate in left or right side of the environment. The process of the approximate columns estimation of the neighbor LEDs is treated as in the case of a single node environment (Figures 4-5). Point one, two or both are repeated for implementing the modified binary search algorithm depending on the locations of the other nodes (Figure 8(a)).

3 The process of the approximate columns estimation of the neighbor LEDs is treated as in the case of a single node environment (Figures 4-5) when all the nodes with sensing LDR sensors less than four have the same number of these sensing LDR sensors.

4 More than one node has a different number of sensing LDR sensors less than four as shown in Fig-

Figure 8

The nodes locations cases on the columns. (a) Different sensing LDRs on four nodes (b) one and two sensing LDRs on node 1 and 2 (c) one and three sensing LDRs on node 1 and 2 (d) two and three sensing LDRs on node 1 and 2

ures 8-9. The process of the approximate columns estimation of the neighbor LEDs is implemented as follows:

− A single LDR sensor on node 1 and two LDR sensors on node 2 sense the   light: the worst case for node 1 location is shown in Figure 9(b). The value of L_{t1} is computed as

$$L_{t1} = N_1 + S, \qquad (9)$$

where N_1 is the distance between the middle lighted column and the left far LDR sensor on node 1. The number of left approximate columns is given by:

$$Lcount1 = L_{t1} / D. \qquad (10)$$

The worst probability for node 2 location is shown in Figure 8(a). The value of R_{t2} is computed as
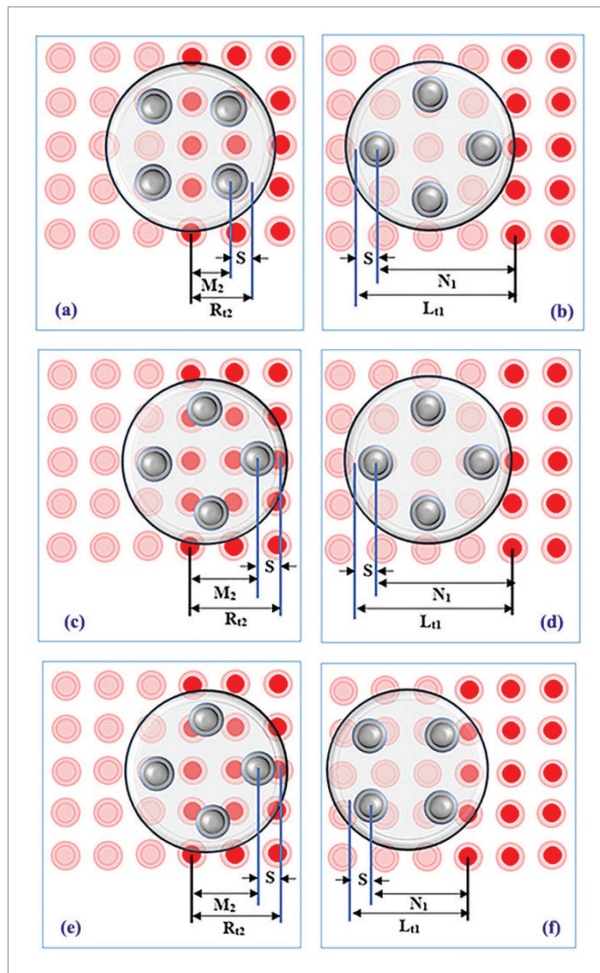
$$R_{t2} = M_2 + S, \qquad (11)$$

where M_2 is the distance between the middle column and the right far LDR sensor on node 2. The number of right approximate columns is:

$$Rcount2 = R_{t2} / D. \tag{12}$$

- A single LDR sensor on node 1 and three LDR sensors on node 2 to sense the LEDs light: The worst probability for node 1 and node 2 locations are shown in Figures 9(d)-(c). The same Equations (9)-(12) can be used to estimate the location.
- Two LDR sensors on node 1 and three LDR sensors on node 2 to sense the LEDs light: The worst probability for node 1 and node 2 locations are shown in Figures 9(e)-(f). The same Equations (9)-(12) can be used to estimate the location of the nodes.

Figure 9

The worst probability for node 1 and 2 locations dependent on the sensitivity of the LDR sensors. (a, b) one and two sensing LDRs (c, d) one and three sensing LDRs (e, f) two and three sensing LDRs for nodes 1 and 2
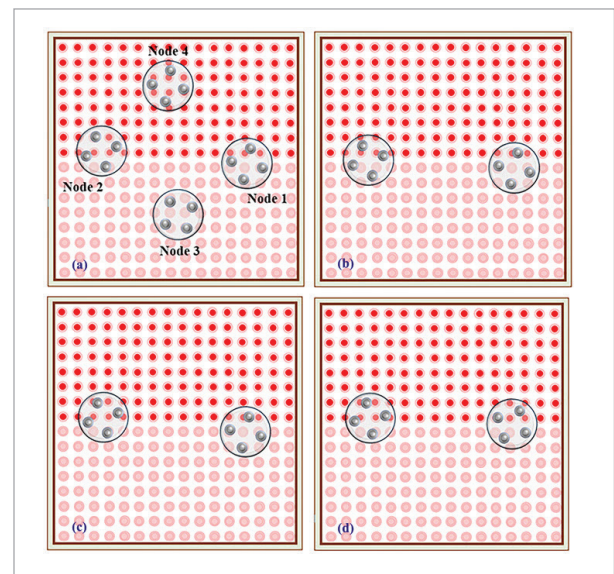

Step 2: Estimate the approximate node rows.

1 In this step, the readings of the LDR sensors are used to distinguish the neighbor rows to each node location. The modified binary search algorithm implemented on the rows of the LED grid. At first, half of these rows are turned on and the others are turned off then the status of the LDR sensors is checked. Figure 10 shows several cases may be obtained depending on the locations of the nodes.

2 All of the LDR sensors of all the nodes may sense the light of the LEDs. This case occurs when all the nodes positions are on the upper part of the environment. This means that the lower part of the environment has no nodes. In this case, the LED rows of the upper half of the upper part are turned on and the rest are turned off. This process is repeated until one of the following cases occurs.

3 One, two or three LDR sensors for one node only sense the light of the LEDs and the other nodes located in the upper or lower side of the environment. The process of the approximate rows estimation of the neighbor LEDs is treated as in the case of a single node environment (Figures 6-7). Point one, two or both are repeated for implementing the modified binary search algorithm depending on the other nodes locations (Figure 10(a)).

Figure 10

Several cases for the nodes locations on the rows. (a) Different sensing LDRs on 4 nodes (b) 1 and 2 sensing LDRs on node 1 and 2 (c) 1 and 3 sensing LDRs on node 1 and 2 (d) 2 and 3 sensing LDRs on node 1 and 2

4 The process of the approximate rows estimation of the neighbor LEDs is treated as in the case of a single node environment (Figures 6-7) when all the nodes with sensing LDR sensors less than four have the same number of these sensing LDR sensors.

5 More than one node has less than four LDR sensors that sense the light as shown in Figures 10-11. The process for estimating the approximate rows of the neighbor LEDs is implemented as follows: A more than one node has less than four LDR sensors that sensing the light as shown in Figures 10-11. The process of the approximate rows estimation of the neighbor LEDs is implemented as follows:

– A single LDR sensor on node 1 and two LDR sensors on node 2 to sense the LEDs light: The worst probability for node 1 location is shown in Figure 11(b). The value of L_{o1} is computed as:

$$L_{o1} = B_1 + S, \qquad (13)$$

where B_1 is the distance between the middle lighted row and the lower far LDR sensor on node 1. The number of lower approximate rows is

$$Locount1 = L_{O1}/D. \qquad (14)$$

The worst probability for node 2 location is shown in Figure 10(a). The value of U_{p2} is computed as:
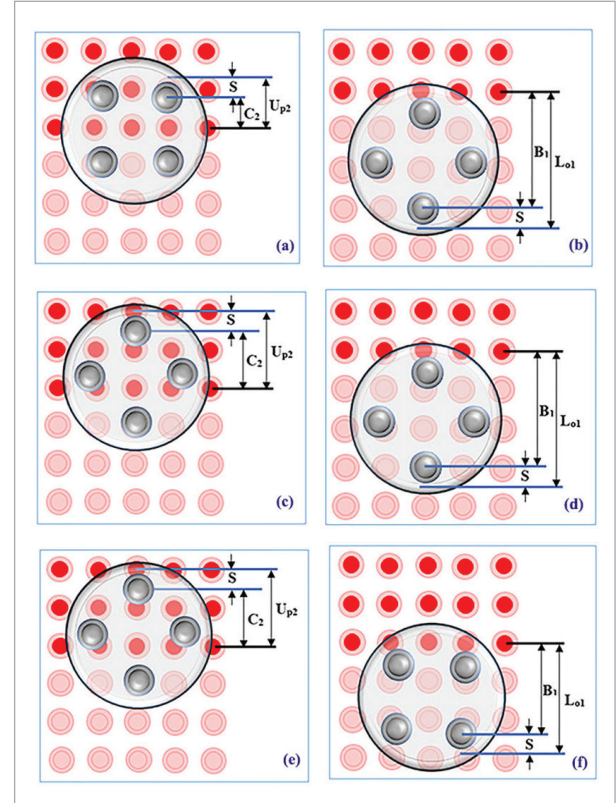
$$U_{p2} = C_2 + S, \qquad (15)$$

where C_2 is the distance between the middle lighted row and the upper far LDR sensor on node 2. The number of upper approximate rows is

$$Upcount2 = U_{p2}/D. \qquad (16)$$

– A single LDR sensor on node 1 and three LDR sensors on node 2 to sense the LEDs light: The worst probability for node 1 and node 2 locations are shown in Figures 11(d)-(c), respectively. The same Equations (13)-(16) can be used in this case to estimate the location of the nodes.

– Two LDR sensors on node 1 and three LDR sensors on node 2 sense the LEDs light: The worst probability for node 1 and node 2 locations are shown in Figures 11(f)-(e) respectively. The same Equations (13)-(16) can be used in this case to estimate the location of the nodes.

Figure 11

The worst probability for node 1 and 2 locations dependent on the sensitivity of the LDR sensors. (a, b) 1 and 2 sensing LDRs (c, d) 1 and 3 sensing LDRs (e,f) 2 and 3 sensing LDRs for nodes 1 and 2

– None of the LDR sensors of all the nodes sense the light of the LEDs. This case occurs when all the nodes positions are in the lower part of the environment. This means that the upper part of the environment has no nodes. In this case, the LED rows of the upper half of the lower part are turned on and the rest are turned off. This process is repeated until one of the above cases occur.

2.2. Accurate Localization Algorithm

This section introduces the proposed second stage that is used to estimate the accurate location of the node by a group of neighbor LEDs. In the previous section, the localization process depends on using the approximate location of the nodes (grid of LEDs as shown in Figure 12(a). In this section, the linear search algorithm is used to estimate the accurate lo-

cation of the node. This algorithm has superiority over the previous algorithm because this algorithm needs less computation when working with small grid of the LEDs. The steps of estimating the accurate location of the nodes are described as follows:

Step 1: Reducing the number of the columns of the approximated LED grid.

The worst case probability of estimating the approximated LED grid gives some columns outside of the node sensing range as shown in Figure 11.b. Since the distance between any two neighbor LDR sensors is R, then the distance between the diagonal LDR is $\sqrt{2}*R$ (worst case to compute the node sensing range). The worst case for the node actual sensing range is computed as:

$$S_n = \sqrt{2}R + 2S. \tag{17}$$

The number of the actual sensing columns for each node is given by:

$$C_{acount} = S_n / D. \tag{18}$$

The number of the approximate columns for the node sensing range is given by:

$$A_{ccount} = L_{count} + R_{count}. \tag{19}$$

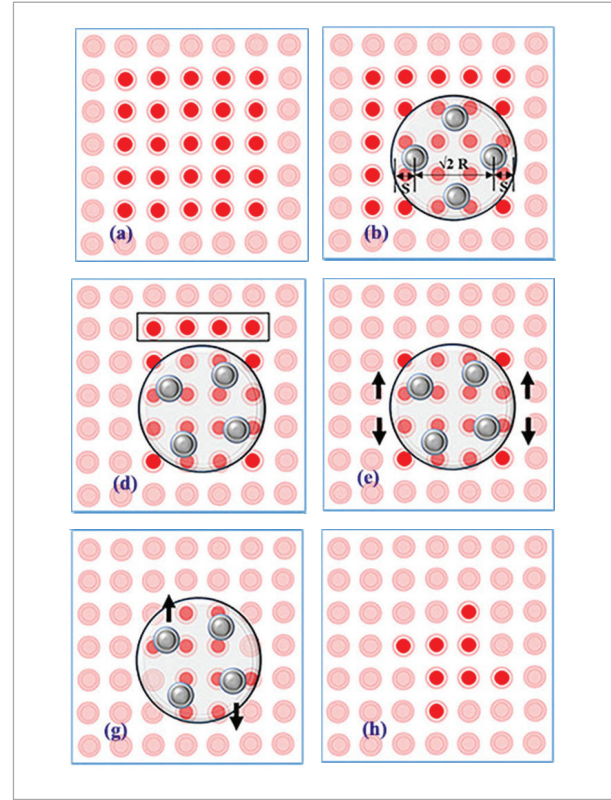$$D_{column} = A_{ccount} - C_{acount}, \tag{20}$$

where D_{column} is the number of the columns that must be disable (disable columns) to give an accurate number of columns to the node sensing range. The linear search algorithm is used to element the disable columns according to the following procedures:

1 Turning on the LEDs in the left column of the approximate columns (Figure 12(c)), then check the status of the node LDR sensors.

2 Repeat step one for other columns until the node LDRs sense the LEDs light.

3 The left checked columns must be removed from the approximate columns.

4 The difference between the number of the removed and the disabled columns must be removed from the right side of the approximate columns.

Step 2: Reducing the number of the rows of the approximated LED grid.

Figure 12

The computing of the node accurate location. (a) The approximate LED grid (b) Removing the non-sensed rows and columns (c) Removing the non-sensed columns (d) Removing the non-sensed rows (e, f and g) Eliminate the non-sensed LEDs (h) The accurate LED grid

The worst case probability of estimating the approximated LED grid gives some rows outside of the node sensing range as shown in Figure 12(b). Since the distance between the diagonal LDR is $\sqrt{2}*R$, then

$$R_{acount} = S_n / D, \tag{21}$$

where R_{acount} is the number of the actual sensing rows for each node.

$$A_{rcount} = U_{pcount} + L_{ocount}, \tag{22}$$

where A_{rcount} is the approximate rows number for the node sensing range.

$$D_{row} = A_{rcount} - R_{acount}, \tag{23}$$

where D_{row} is the number of the rows that must be disable (disable rows) to give an accurate number of rows to the node sensing range. The linear search algorithm is used to element the disable rows according to the following procedures:

1   Turn on the LEDs in the upper row of the approximate columns (Figure 12(d)) then check the status of the node LDR sensors.

2   Repeat the previous step for the other down rows until the node LDRs sense the LEDs light.

3   The upper checked columns must be removed from the approximate rows.

4   The difference between the number of the removed and the disable rows must be removed from the lower side of the approximate rows.

Step 3: Enhancement the accurate LED grid by removing the non-sensed LEDs.

A modified linear search algorithm is suggested here to complete the accurate LED grid by removing the non-sensed LEDs. The linear search algorithm is modified to work with two dimensional binary numbers that are represented by the status of the LEDs. The LED grid is divided into four quarters and the following procedures are used to treat with all of these quarters:

1   Start from the outer column to inner one for each quarter of the LED grid.

2   Start from inner LED to an outer one in current column as in Figure 12(e).

3   Turn on the LED and check the status of the node LDR sensors.

4   Repeat linearly the third step for other upper LEDs until the node LDRs not sense the LEDs light (Figure 12(f)).

5   The lower checked LEDs must be fixed into the accurate LED grid and the other are removed.

6   Move one column and repeat the last three points as shown in Figure 12(g).

7   The procedure is repeated until finalizing the quarters of the LED grid (Figure 12(h)). The procedure repeating until finalizing the quarters of the LED grid (Figure 12(h)).

2.3. Multi-Node Localization Algorithm

This section explains two algorithms: Minimum Bounded Circle of 2D Convex Hull and Centroid algorithms both are used to estimate the location and identification of the multi-nodes using the accurate LED grid which represent the LEDs within the sensing range of the nodes. The identification process is simply solved since each node sends its ID with information collected by its LDR sensors. The localization process of the multi-node system is similar to the case of a single node since each node is manipulated alone. The Minimum Bounded Circle of 2D Convex Hull algorithm localizes any node by representing the LED grid as a polygon convex hull and then computing the circle with minimum boundary. The center of this circle represents the node location. In centroid approach, the LED grid is treated as a polygon and the centroid (the geometric center) is the arithmetic mean position of all the points in the polygon.

2.3.1. Minimum Bounded Circle of the 2D Convex Hull Algorithm

This algorithm is used to compute the polygon convex hull from a set of points that is used to obtain a minimum bounded circle. The convex hull is obtained by choosing the smallest subset of the points which does not include any of the inner set points. Chan's algorithm is one of the best known algorithms used to build the convex hull of set points in a 2D or 3D environment [2]. Chan's algorithm is based on using both the Jarvis March algorithms and Graham's scan. In this section, a new algorithm "maxima boundaries convex hull algorithm" for polygon convex hull construction is introduced instead of Chan's algorithm.
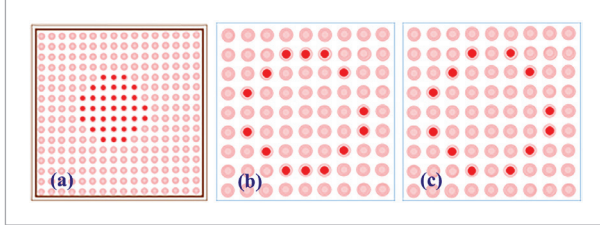
A. Maxima Boundaries Convex Hull Algorithm

This algorithm produces a new method to construct a polygon convex hull from a set of point represented by the LED grid. The process of the construction depends on scanning 2D array with the coordinate axis of the entire LED grid where the coordinates with maximum and minimum values are chosen to build the convex hull. The steps of implementing this algorithm are as follows:

1   Store the coordinate axis of the LED grid (red LED in Figure 13(a)) in two dimensional array, the first index of each element in the array represents the LED number and the second one is used to store the x and y coordinate axis of each LED.

2   Use the linear search algorithm to classify the elements of the array into groups, each one has the same value of the x coordinate axis. Store the elements with maximum and minimum y value in each group in a

Figure 13

The steps of applying the maxima boundaries convex hull algorithm on the accurate location LED grid (a) The accurate location LED grid (b) Create the elements of the boundary convex hull array (c) Improved the elements of the boundary convex hull array

new grid to represent the elements of the boundary convex hull array. Remove the reminder element of the old array (Figure 13(b)).

3 Improve the elements of the boundary convex hull array by removing some LEDs with inner values. This process is achieved by using the linear search algorithm to search and classify the elements of the boundary convex hull array into groups; each one has the same value of the y coordinate axis. Store the element with the maximum and the minimum x value in each group in the same grid to represent the elements of the enhanced boundary convex hull array. Then, the unused elements from this array are removed (Figure 13(c)).

B. The Minimum Bounded Circle Algorithm

The node localization process is achieved by applying the minimum bounded circle algorithm. This algorithm is used to draw a minimum bounded circle around the boundary convex hull elements which are obtained from the last algorithm and the center of this circle represents the node location. The steps for implementing this algorithm are described below:

1 Choose the line with the maximum length between any two elements in the boundary convex hull elements as the initial value of circle diameter and the center of the diameter as the circle center.

2 Check the distances among all the boundary convex hull elements and the center of the new circle. Choose the longest one and extended it to intersect the far side of the circle. This new line is used as the diameter of the new circle.

3 The last step is repeated until all the boundary convex hull elements have distances less than the ra-

dius of the last tested circle. This circle is the minimum bounded circle.

4 The midpoint of the last circle diameter represents the position of the node.

2.3.2. The Centroid Algorithm

In the centroid approach, the accurate LED grid is treated as a polygon and the centroid (the geometric center) is the arithmetic mean position of all the points in the polygon. When the elements of the accurate LED grid treated as the vertices (x_o, y_o), (x_1, y_1)... (x_{n-1}, y_{n-1}), the coordinates of the centroid point are calculated as:

$$C_x = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \tag{24}$$

$$C_y = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \tag{25}$$

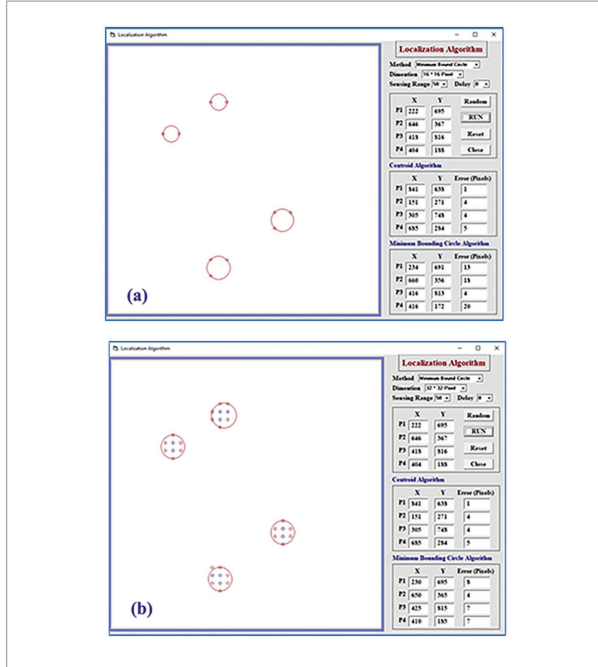$$A = \frac{1}{2} \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i). \tag{26}$$

3. Simulation Results

The multi-beacon searching algorithm is simulated by using the Visual Basic dot Net software. Simulations are executed over 100 different topologies, different LED grid and different number of nodes. The nodes are randomly placed in the environment with 16×16, 32×32 and 64×64 LED grid. For each topology, the LDRs sensing range is varied in order to study the full detection range and the percentage of the error that occurred during the localization process. The system parameters that used in this simulation are:
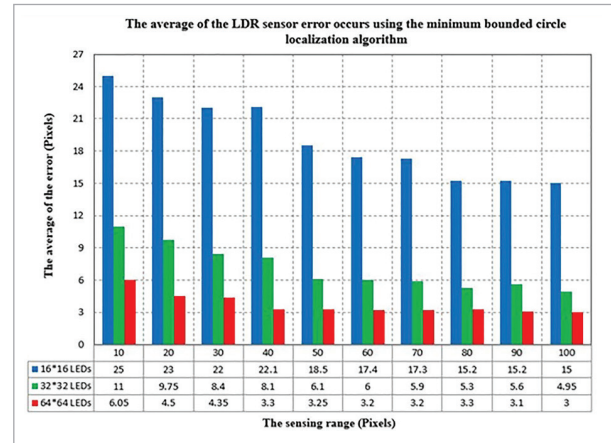
1 The environment size n: the number of LEDs in the environment (n*n LEDs). Changing the network size will indirectly change the node localization accuracy.

2 Maximum sensing range of LDR sensors S: this parameter affects the full detection range and the accuracy in the localization process.

3 Number of LDR sensors m: this parameter affects the time of execution.
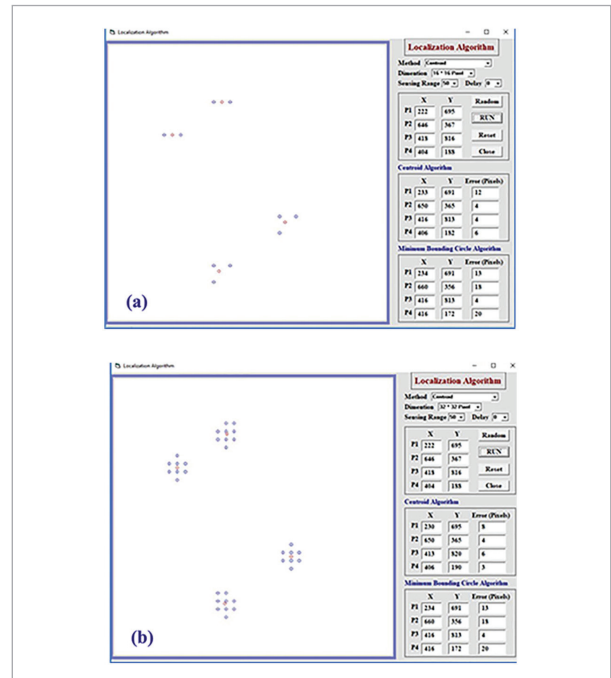
Figure 14

The using of the minimum bounded circle algorithm with LDRs have 50 Pixels sensing range (a) The 16´16 LED grid (b) The 32´32 LED grid

Figure 15

A comparison between the average of the error and the sensing range of the LDR sensors using centroid localization algorithm in (16×16), (32×32) and (64×64) LED grid environment

The average of the LDR sensor error occurs using the minimum bounded circle localization algorithm

	10	20	30	40	50	60	70	80	90	100
16*16 LEDs	25	23	22	22.1	18.5	17.4	17.3	15.2	15.2	15
32*32 LEDs	11	9.75	8.4	8.1	6.1	6	5.9	5.3	5.6	4.95
64*64 LEDs	6.05	4.5	4.35	3.3	3.25	3.2	3.2	3.3	3.1	3

The sensing range (Pixels)

Figure 16

The applying of the centroid algorithm with LDR sensors have 50 Pixels sensing range (a) The 16×16 LED grid (b) The 32×32 LED grid
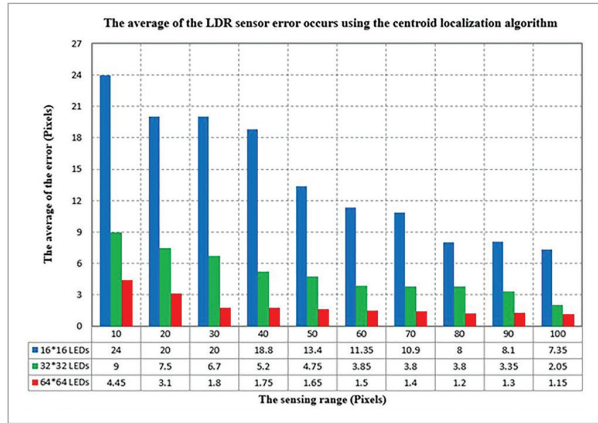

At first, the effect of changing the LDRs sensing range on the average of the error that occurs in the localization process is illustrated using the minimum bounded algorithm. Figure 14 shows the simulation executed on 16×16 and 32×32 LED grid with LDR sensors have 50 Pixels sensing range. Figure 15 demonstrates the comparison between the average of the detecting neighbor LEDs and the sensing range in 16×16, 32×32 and 64×64 LED grid.

The second simulation is used to study the impact of changing the sensing range of the LDR sensors on the average of the error occurs in the node localization using the centroid algorithm. Figure 16 shows the simulation executed on all the LED grid environment with LDR sensors have 50 Pixels sensing range. Figure 17 demonstrates the comparison between the average of the detecting neighbor LEDs and the sensing range of the LDR sensors in (16×16), (32×32) and (64×64) LED grid. The results in the last two simulations demonstrate that when the sensing range is increased, the average error will be decreased and the best case happens when the environment has (64×64) LEDs. The centroid approach has more accurate reading than the minimum bounded circle algorithm.
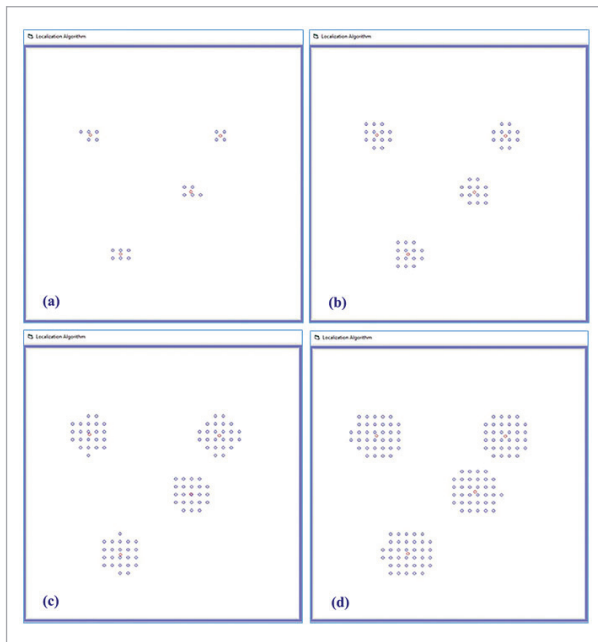
Figure 17

A comparison between the average of the error and the sensing range of the LDR sensors using the centroid localization algorithm in (16×16), (32×32) and (64×64) LED grid

A further example is intended to show the effect of changing the LDRs sensing range on the percentage of the full detecting range. Figure 18 shows the simulation executed on 32×32 LEDs environment with
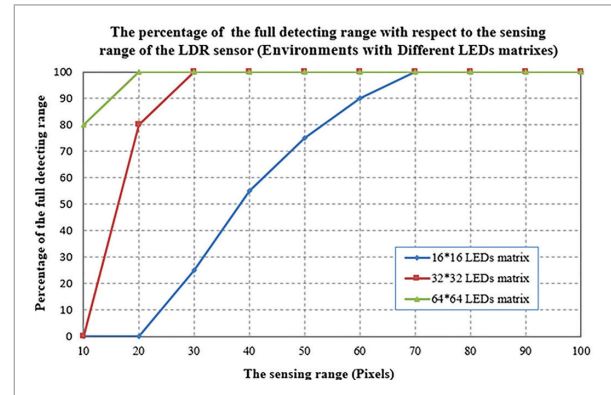
Figure 18

A simulation for different cases of the detected neighbors LEDs in 32×32 LEDs environment (a) 40 Pixels sensing range (a) 60 Pixels sensing range (a) 80 Pixels sensing range (a) 100 Pixels sensing range

LDR sensors have 40, 60, 80 and 100 Pixels sensing range. Figure 19 shows the comparison between the percentage of the full detecting range of the neighbor LEDs and the sensing range of the LDR sensors in (16×16), (32×32) and (64×64) LED grid environment.

Figure 19

A comparison between the percentage of the full detecting range of the neighbor LEDs and the LDRs sensing range in (16×16), (32×32) and (64×64) LEDs environment

The result shows that when the sensing range is increased, then the percentage of the full detecting range of the neighbor LEDs will increased too. The full detecting range of the neighbor LEDs is 70 Pixels for (16×16) LEDs, 30 Pixels for (32×32) LEDs and 20 Pixels for (32×32) LEDs. The best case for the full detecting range of the neighbor LEDs occurs when the environment has (64×64) LEDs.

The last simulation results are used to study the effect of changing the sensing range of the LDR sensors on the execution time for the localization process in different sizes of LED grid environment and different number of nodes for each simulation process. Figure 20 shows the simulation executed on 64×64 LED grid environment with one, two, three and four nodes localization process.

Figure 21 shows the comparison between the execution time of the one node localization process and the sensing range of the LDR sensors in (16×16), (32×32) and (64×64) LED grid environment. The execution time is affected by the speed of the computer and the software written for this purpose. Since all the simulations are executed on the same computer and the same software, the term (per unit time) is used as a unit to the time of execution.

Figure 20

The simulation shows the execution speed with different cases for the nodes number in 64×64 LED grid environment (a) one node (a) two nodes (a) three nodes (a) four nodes
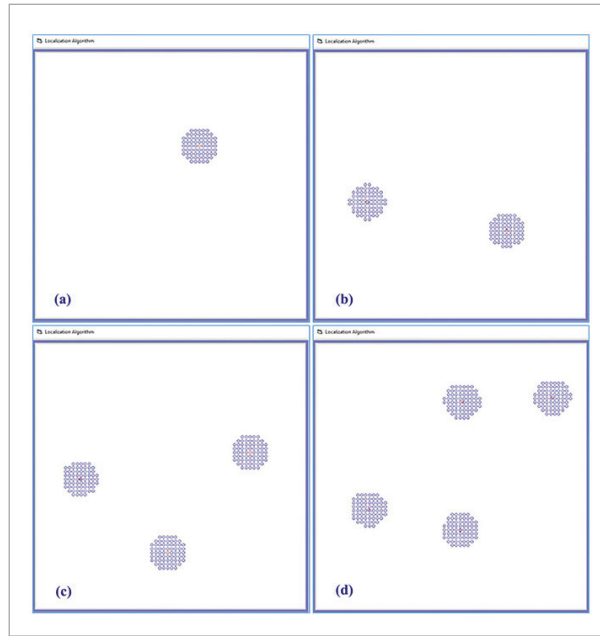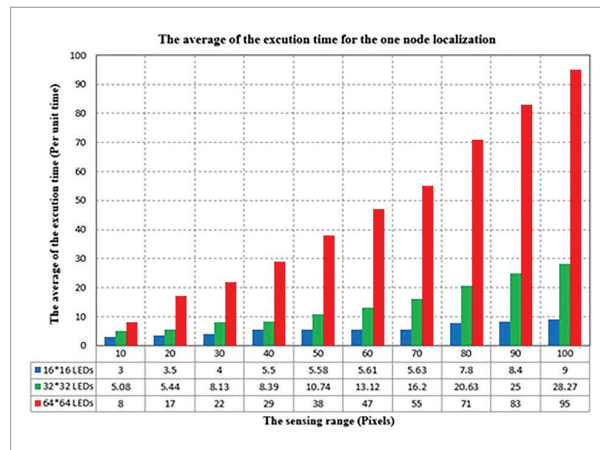

Figure 21

A comparison between the average of the execution time for the one node localization process and the sensing range of the LDR sensors in all the LED grid environment

Figures 22-24 show the same comparison as in Figure 20. These comparisons are done between the execution time of the two, three and four nodes localization process, respectively, and the sensing range of the

LDR sensors in (16×16), (32×32) and (64×64) LED grid environment.

The result shows that, as the sensing range is increased, the average of execution time is decreased for all types of LED grid environment and in one, two, three and four nodes localization process. When the number of the neighbor LEDs for each node is increased, the sensing range is increased and this leads to increasing the time of searching by the modified bi-

Figure 22

A comparison between the average of the execution time for the two nodes localization process and the sensing range of the LDR sensors in (16×16), (32×32) and (64×64) LED grid environment
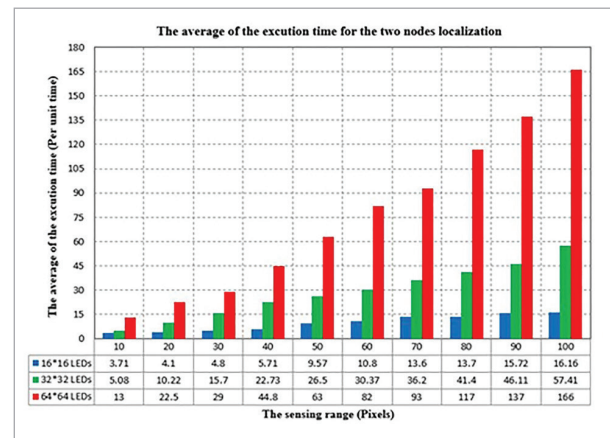

Figure 23

A comparison between the average of the execution time for the three nodes localization process and the sensing range of the LDR sensors in (16×16), (32×32) and (64×64) LED grid environment
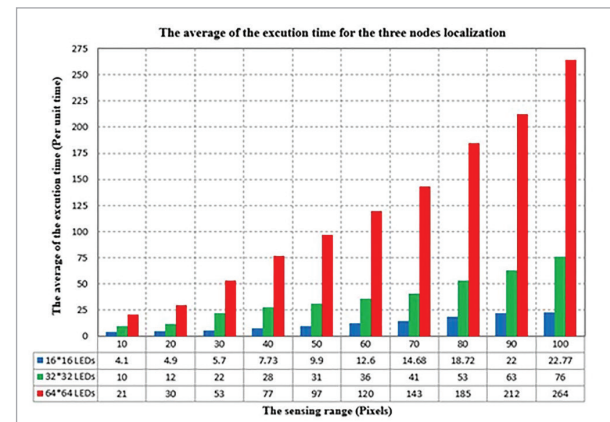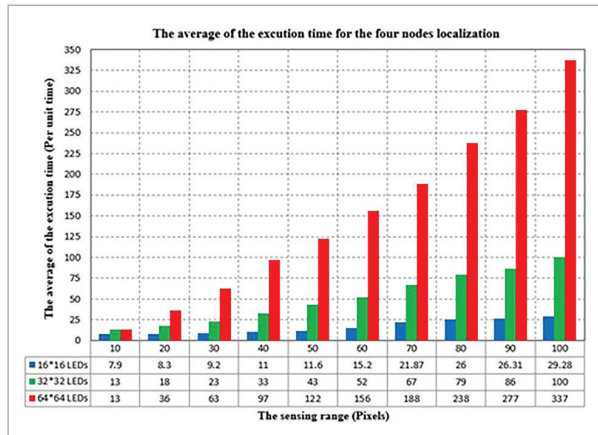
Figure 24

A comparison between the average of the execution time for the four nodes localization process and the sensing range of the LDR sensors in (16×16), (32×32) and (64×64) LED grid environment

nary search algorithm. The best case of the execution time of the localization process occurs when the environment has 16×16 LED grid, but this type of environment produces a less accuracy in the localization process. Figures 21-24 show that the one node localization has less execution time since in this process the number of the neighbor LEDs of the nodes in the localization process are reduced.

4. Conclusion

In this paper, a new multi-beacon searching algorithm for multi-node localization and identification has been introduced. This algorithm depends on searching in two-dimensional LED grid by using LDR sensors placed on nodes. The collected information by LDR has been used for the localization process based on either the minimum bounded circle or the centroid algorithms.

The proposed multi-node localization system has a low cost. It only needs LEDs and LDR sensors for the

localization process while the problem of a limited number of nodes has been solved because of no need for node ID coding. Simulation results show that when the sensing range is incremented, the average of error in the localization process will be decreased. The best case happens when the environment has (64×64) LED grid for all types of the environment (16×16, 32×32 and 64×64 LED grid). In addition, it is observed that the centroid approach has more accurate reading than the minimum bounded circle algorithm for both the value of the sensing range and all the types of the environment.

The full detecting range of the neighbor LEDs is a significant factor since it indicates the LDR sensing range that makes the LED grid always visible by the LDR sensors. From the simulation results, it is found that as the LDR sensing range is increased, the percentage of the full detecting range of the neighbor LEDs will be increased too. The full detecting range of the neighbor LEDs is 70 Pixels for (16×16) LED grid, 30 Pixels for (32×32) LED grid and 20 Pixels for (32×32) LED grid. The best case for the full detecting range of the neighbor LEDs occurs when the environment has (64×64) LED grid.

The execution time of the node localization is an important factor because it shows the validation of the proposed algorithms for localization in a dynamic environment. The simulation results show that as the sensing range of the LDR sensors is increased, the average of execution time is decreased for all types of the LED grid of one, two, three and four nodes localization process. When the number of the neighbor LEDs for each node is increased, the sensing range will be increased and the time of searching by the modified binary search algorithm is increased, too. The best case of the execution time of the localization process occurs when the environment has 16×16 LED grid, but this type of environment produces less accuracy in the localization process. In addition, the one node localization has less execution time since in this process the number of the neighbor LEDs of the nodes is reduced.

References

1. Bae, J., Lee, S., Song, J. Use of coded infrared light for mobile robot localization, Journal of Mechanical Science and Technology, 2008, 22(7), 1279-1286. https://doi.org/10.1186/s12984-016-0151-8

2. Chan, T. M. Optimal Output-Sensitive Convex Hull Algorithms in Two and Three Dimensions. Discrete & Computational Geometry, 1996, 16(4), 361-368. https://doi.org/10.1007/BF02712873

3. Chen, C., Kai-Tai, S. Complete Coverage Motion Control of a Cleaning Robot Using Infrared Sensors. IEEE international conference on mechatronics, 2005, 543-548.

4. Desouza, G. N., Kak, A. C. Vision for mobile robot navigation: a survey, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24 (2), 237-267. https://doi.org/10.1109/34.982903

5. Galvan, C. E., Galvan-Tejada, I., Sandoval, E. I., Brena, R. Wifi Bluetooth Based Combined Positioning Algorithm. Procedia Engineering, 2012, 35, 101-108. https://doi.org/10.1016/j.proeng.2012.04.170

6. Gorostiza, E. M., Galilea, J. L. L., Meca, F. J. M., Monzú, D. S., Zapata, F. E., Puerto, L. P. Infrared sensor system for mobile-robot positioning in intelligent spaces, Sensors, 2011, 11(5), 5416-5438. https://doi.org/10.3390/s110505416

7. Han, G., Jiang, J., Shu, L., Xu, Y., Wang, F. Localization algorithms of underwater wireless sensor networks: A survey, Sensors, 2012, 12, 2026-2061. https://doi.org/10.3390/s120202026

8. Jekabsons, G., Kairish, V., Zuravlyov, V. An analysis of wi-fi based indoor positioning accuracy, Applied Computer Systems, 2011, 44(1), 131-137. https://doi.org/10.2478/v10143-011-0031-4

9. Koyuncu, H., Yang, S. H. A survey of indoor positioning and object locating systems, International Journal of Computer Science and Network Security, 2010, 10(5), 121-128.

10. Krejsa, J., Vechet, S. Infrared beacons based localization of mobile robot, Electronics and Electrical Engineering, 2012, 117(1), 17-22. https://doi.org/10.5755/j01.eee.117.1.1046

11. Lee, J., Hyun, CH., Park, M. A vision-based automated guided vehicle system with marker recognition for indoor use, Sensors, 2013, 13(8), 10052-10073. https://doi.org/10.3390/s130810052

12. Lee, S. Use of infrared light reflecting landmarks for localization, Industrial Robot, 2009, 36(2), 138-145. https://doi.org/10.1108/01439910910932595

13. Lee, S., Song, J. Use of coded infrared light as artificial landmarks for mobile robot localization, 2007 IEEE/RSJ international conference on intelligent robots and systems, 2007, 1731-1736. https://doi.org/10.1109/IROS.2007.4399600

14. Leung, K. Y. K, Barfoot, T. D., Liu, H. H. T. Decentralized localization of sparsely communicating robot networks: a centralized-equivalent approach, IEEE Transactions on Robotics, 2010, 26(1), 62-77. https://doi.org/10.1109/TRO.2009.2035741

15. Liu, H., Darabi, H., Banerjee, P., Liu, J. Survey of wireless indoor positioning techniques and systems, IEEE Transactions on Systems, Man, and Cybernetics, Part C, 2007, 37(6), 1067-1080. https://doi.org/10.1109/TSMCC.2007.905750

16. Luo, P., Zhang, M., Zhang, X., Cai, G., Han, D., Li, Q. An indoor visible light communication positioning system using dual-tone multi-frequency technique, International workshop on optical wireless communications (IWOW), 2013. https://doi.org/10.1109/IWOW.2013.6777770

17. Mao, L., Chen, J., Li, Z., Zhang, D. Relative localization method of multiple micro robots based on simple sensors, International Journal of Advanced Robotic Systems, 2013, 10(2), 1-9. https://doi.org/10.5772/55587

18. Marjovi, A., Marques, L. Multi-Robot Olfactory Search in Structured Environments. Robotics and Autonomous Systems, 2011, 52, 867-881. https://doi.org/10.1016/j.robot.2011.07.010

19. Mi, J., Takahashi, Y. Performance analysis of mobile robot self-localization based on different configurations of rfid system, 2015 IEEE international conference on advanced intelligent mechatronics (AIM), July 7-11, 2015, Busan, Korea, 2015, 1591-1596. https://doi.org/10.1109/AIM.2015.7222770

20. Odokuma, E. E., Owolabi, O. An indexed method for improving the efficiency of the binary search algorithm, International Journal of Advanced Research in Computer Science and Software Engineering, 2016, 6, 116-122.

21. Oh, J.H., Kim, D., Lee, B. H. An indoor localization system for mobile robots using an active infrared positioning sensor, Journal of Industrial and Intelligent Information, 2014, 2(1), 35-38. https://doi.org/10.12720/jiii.2.1.35-38

22. Rashid, A. T., Frasca, M., Ali, A. A., Rizzod, A., Fortuna, L. Multi-Robot Localization and Orientation Estimation Using Robotic Cluster Matching Algorithm. Robotics and Autonomous Systems, 2015, 63 (1), 108-121. https://doi.org/10.1016/j.robot.2014.09.002

23. Ul-Haque, I., Prassler, E. Experimental evaluation of a low-cost mobile robot localization technique for large indoor public environments, ISR 2010 (41st international symposium on robotics) and ROBOTIK 2010 (6th German conference on robotics), 2010, 1-7.

24. Wang, J., Takahashi, Y. Indoor mobile robot self-localization based on a low-cost light system with a novel emitter arrangement, Robomech Journal, 2018, 5(17), 1-17. https://doi.org/10.1186/s40648-018-0114-x

25. Wang, Z., Hwang, Y., Kim, Y., Lee, D., Lee, J. Mobile robot indoor localization using surf algorithm based on lrf sensor, 54th annual conference of the society of instrument and control engineers of Japan (SICE), 2015, 1122-1125. https://doi.org/10.1109/SICE.2015.7285500

26. Yayan, U., Yucel, H., Yazıcı, A. A low cost ultrasonic based positioning system for the indoor navigation of mobile robots, Journal of Intelligent & Robotic Systems, 2015, 78(3), 541-552. https://doi.org/10.1007/s10846-014-0060-7

27. Zhang, S., Cao, Y. Cooperative localization approach for multi-robot systems based on state estimation error compensation, Sensors, 2019, 19(18), 38-42. https://doi.org/10.3390/s19183842